UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/899,539	07/06/2001	David S. Ebbo	40062.0265US01	3399

27488 7590 12/09/2008

MERCHANT & GOULD (MICROSOFT)
P.O. BOX 2903
MINNEAPOLIS, MN 55402-0903

EXAMINER	
AILES, BENJAMIN A	

ART UNIT	PAPER NUMBER
2442	

MAIL DATE	DELIVERY MODE
12/09/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	09/899,539	EBBO ET AL.	
	Examiner	Art Unit	
	BENJAMIN AILES	2442	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *30 September 2008*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *14-40* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *14-40* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date. _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 14-40 remain pending.

Continued Examination Under 37 CFR 1.114

2. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 03 July 2008 has been entered.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 14-18, 27, 28, and 30-34 are rejected under 35 U.S.C. 103(a) as being unpatentable over Tiemann et al. (US 7,171,443 B2), hereinafter referred to as Tiemann, in view of Tso et al. (US 6,892,226 B1), hereinafter referred to as Tso, and further in view of Rana (US 2005/0027823 A1).

5. Regarding claim 14, Tiemann teaches a machine-readable medium having instructions recorded thereon, such that when the instructions are read and executed by a processor in a first computing system connected to a network, the first computing system performs a method comprising: receiving, at the first computing system, a

request for a web page from a second computing system, the request web page having content (col. 2, ll. 26-28), creating on the first computing system a page object having references to component objects in response to the received request for information, the page object being created based on a page file, each component object of the page object representing a user control within the page file (col. 6, ll. 1-10; use of a template). Tiemann teaches the identification of data objects ("determining whether each object referenced by the page object corresponds to a user control; determining whether each user control supports output caching; determining whether the object referenced by the page object is cached if the object corresponds to a user control that supports output caching") and retrieving the output data of the object referenced by the page object and retrieving executable code for objects (col. 6, ll. 1-10, Tiemann teaches the identification of dynamic and static portions; col. 7, ll. 9-15, use of JavaScript) wherein certain objects are identified as cached objects (col. 6, ll. 6-10) and the use of a cache key that comprises an identifier that is unique for each occurrence of the component object (col. 7, ll. 60-67, filenames for cached objects created based on user identifiers and time stamps). Tiemann teaches further the inserting any retrieved output data and any created objects as components into a hierarchical tree data model at the server computing system (col. 6, ll. 1-37, use of template file and example template file), processing the components of the hierarchical tree data model at the server computing system to create a renderable page object (col. 7, ll. 4-11, after portions of the template have been identified the HTML file is generated); and sending the renderable page object from the server computing system to the client computing system (col. 2, ll. 32-

37). Tiemann teaches the use of a cache that is locally accessible by a user (col. 4, ll. 25-27) however (a) does not teach of a cache that is located at the server. Further, Tiemann (b) does not explicitly teach that in the hierarchical tree data model that "each component object being linked to a prior component object if a prior component object exists, and each component object being linked to a next component object if a next component object exists.

(a) In related art, Tso teaches similarly to Tiemann the request by a client being sent to a server for information (col. 3, ll. 14-19) and Tso teaches further the utilization of retrieving content located at the server and storing and retrieving the content in and from a server-side cache memory (col. 4, line 65 – col. 5, line 11). One of ordinary skill in the art at the time of the applicants' invention would have recognized the use of a server side cache memory and therefore would have found it obvious in combination with Tiemann to utilize a cache memory as taught by Tso. One of ordinary skill in the art would have been motivated to utilize a server-side cache memory as taught by Tso wherein Tso teaches that it is advantageous to utilize a cache memory when needing to retrieve content for a user and have versions of content for later use without the need to re-retrieve content from a network source (Tso, col. 5, ll. 8-11).

(b) With respect to limitation (b), Tiemann teaches on the aspect of a hierarchical tree data model with respect to a plurality of component objects organized in markup language in a template file (col. 6, ll. 1-37) however does not explicitly teach the component object linking. However, in related art, Rana teaches on this aspect of the invention wherein Rana teaches the common usage of tree data models with respect to

markup language formatting (abstract and p. 1 para. 0012). Rana teaches the linking of component objects wherein the processing of a document object model (DOM) is performed, the portions of the tree are decoded with respect to positional information and other page items and displayed to a user by appropriate means. One of ordinary skill in the art at the time of the applicants' invention would have recognized the common use of a tree data model and therefore would have found it obvious in combination with Tiemann to utilize a tree data model as taught by Rana. One of ordinary skill in the art would have been motivated to utilize a document object model tree as taught by Rana wherein Rana teaches the common utilization of a DOM tree and view tree that includes all Javascript for the client (Rana, p. 1, para. 0017).

6. Regarding claim 15, Tiemann, Tso and Rana teach wherein: the user control including an output caching directive, wherein caching the component object in the output cache comprises caching the component object according to the output caching directive (Tso, col. 6, ll. 14-16; Tiemann, col. 6, ll. 39-44, use of a CACHE tag).

7. Regarding claim 16, Tiemann, Tso and Rana teach the method wherein the contents of the renderable page comprises an HTML specification for a web page (Tiemann, col. 7, ll. 4-9).

8. Regarding claim 17, Tiemann, Tso and Rana teach the method wherein: the step of processing the created objects comprises processing each one of the components individually (Tiemann, col. 6, ll. 51-57).

9. Regarding claim 18, Tiemann, Tso and Rana teach the method further comprising:

creating the hierarchical tree data model including each of the components and a hierarchical relationship among the components, the data model being used during the step of processing the page to facilitate processing each of the components (Tiemann, col. 6, ll. 1-37, use of template file and example template file).

10. Regarding claim 27, Tiemann teaches a method for providing a response to a request for information from a client computing system comprising receiving, at the server computing system, a request for information from the client computing system (col. 2, ll. 26-28), creating a page object having references to objects (col. 6, ll. 1-10; use of a template). Tiemann teaches the identification of data objects ("determining whether each object referenced by the page object corresponds to a user control; determining whether each user control supports output caching; determining whether the object referenced by the page object is cached if the object corresponds to a user control that supports output caching") and retrieving the output data of the object referenced by the page object and retrieving executable code for objects (col. 6, ll. 1-10, Tiemann teaches the identification of dynamic and static portions; col. 7, ll. 9-15, use of JavaScript) wherein certain objects are identified as cached objects (col. 6, ll. 6-10). Tiemann teaches further the inserting any retrieved output data and any created objects as components into a hierarchical tree data model at the server computing system (col. 6, ll. 1-37, use of template file and example template file), processing the components of the hierarchical tree data model at the server computing system to create a renderable page object (col. 7, ll. 4-11, after portions of the template have been identified the HTML file is generated); and sending the renderable page object from the server computing

system to the client computing system (col. 2, ll. 32-37). Tiemann teaches the use of a cache that is locally accessible by a user (col. 4, ll. 25-27) however (a) does not teach of a cache that is located at the server. Further, Tiemann (b) does not explicitly teach that in the hierarchical tree data model that "each component object being linked to a prior component object if a prior component object exists, and each component object being linked to a next component object if a next component object exists.

(a) In related art, Tso teaches similarly to Tiemann the request by a client being sent to a server for information (col. 3, ll. 14-19) and Tso teaches further the utilization of retrieving content located at the server and storing and retrieving the content in and from a server-side cache memory (col. 4, line 65 – col. 5, line 11). One of ordinary skill in the art at the time of the applicants' invention would have recognized the use of a server side cache memory and therefore would have found it obvious in combination with Tiemann to utilize a cache memory as taught by Tso. One of ordinary skill in the art would have been motivated to utilize a server-side cache memory as taught by Tso wherein Tso teaches that it is advantageous to utilize a cache memory when needing to retrieve content for a user and have versions of content for later use without the need to re-retrieve content from a network source (Tso, col. 5, ll. 8-11).

(b) With respect to limitation (b), Tiemann teaches on the aspect of a hierarchical tree data model with respect to a plurality of component objects organized in markup language in a template file (col. 6, ll. 1-37) however does not explicitly teach the component object linking. However, in related art, Rana teaches on this aspect of the invention wherein Rana teaches the common usage of tree data models with respect to

markup language formatting (abstract and p. 1 para. 0012). Rana teaches the linking of component objects wherein the processing of a document object model (DOM) is performed, the portions of the tree are decoded with respect to positional information and other page items and displayed to a user by appropriate means. One of ordinary skill in the art at the time of the applicants' invention would have recognized the common use of a tree data model and therefore would have found it obvious in combination with Tiemann to utilize a tree data model as taught by Rana. One of ordinary skill in the art would have been motivated to utilize a document object model tree as taught by Rana wherein Rana teaches the common utilization of a DOM tree and view tree that includes all Javascript for the client (Rana, p. 1, para. 0017).

11. Regarding claim 28, Tiemann teaches a method for providing a response to a request for information from a client computing system comprising receiving, at the server computing system, a request for information from the client computing system (col. 2, ll. 26-28), creating a page object having references to objects (col. 6, ll. 1-10; use of a template). Tiemann teaches the identification of data objects ("determining whether each object referenced by the page object corresponds to a user control; determining whether each user control supports output caching; determining whether the object referenced by the page object is cached if the object corresponds to a user control that supports output caching") and retrieving the output data of the object referenced by the page object and retrieving executable code for objects (col. 6, ll. 1-10, Tiemann teaches the identification of dynamic and static portions; col. 7, ll. 9-15, use of JavaScript) wherein certain objects are identified as cached objects (col. 6, ll. 6-10).

Tiemann teaches further the inserting any retrieved output data and any created objects as components into a hierarchical tree data model at the server computing system (col. 6, ll. 1-37, use of template file and example template file), processing the components of the hierarchical tree data model at the server computing system to create a renderable page object (col. 7, ll. 4-11, after portions of the template have been identified the HTML file is generated); and sending the renderable page object from the server computing system to the client computing system (col. 2, ll. 32-37). Tiemann teaches the use of a cache that is locally accessible by a user (col. 4, ll. 25-27) however (a) does not teach of a cache that is located at the server. Further, Tiemann (b) does not explicitly teach that in the hierarchical tree data model that "each component object being linked to a prior component object if a prior component object exists, and each component object being linked to a next component object if a next component object exists.

(a) In related art, Tso teaches similarly to Tiemann the request by a client being sent to a server for information (col. 3, ll. 14-19) and Tso teaches further the utilization of retrieving content located at the server and storing and retrieving the content in and from a server-side cache memory (col. 4, line 65 – col. 5, line 11). One of ordinary skill in the art at the time of the applicants' invention would have recognized the use of a server side cache memory and therefore would have found it obvious in combination with Tiemann to utilize a cache memory as taught by Tso. One of ordinary skill in the art would have been motivated to utilize a server-side cache memory as taught by Tso wherein Tso teaches that it is advantageous to utilize a cache memory when needing to

retrieve content for a user and have versions of content for later use without the need to re-retrieve content from a network source (Tso, col. 5, ll. 8-11). .

(b) With respect to limitation (b), Tiemann teaches on the aspect of a hierarchical tree data model with respect to a plurality of component objects organized in markup language in a template file (col. 6, ll. 1-37) however does not explicitly teach the component object linking. However, in related art, Rana teaches on this aspect of the invention wherein Rana teaches the common usage of tree data models with respect to markup language formatting (abstract and p. 1 para. 0012). Rana teaches the linking of component objects wherein the processing of a document object model (DOM) is performed, the portions of the tree are decoded with respect to positional information and other page items and displayed to a user by appropriate means. One of ordinary skill in the art at the time of the applicants' invention would have recognized the common use of a tree data model and therefore would have found it obvious in combination with Tiemann to utilize a tree data model as taught by Rana. One of ordinary skill in the art would have been motivated to utilize a document object model tree as taught by Rana wherein Rana teaches the common utilization of a DOM tree and view tree that includes all Javascript for the client (Rana, p. 1, para. 0017).

12. Regarding claim 30, Tiemann, Tso and Rana teach the method wherein the contents of the renderable page comprises an HTML specification for a web page (Tiemann, col. 7, ll. 4-9).

13. Regarding claim 31, Tiemann, Tso and Rana teach the method further comprising arranging the page components into a data model to facilitate rendering the

requested web page based on the page components (Tiemann, col. 7, ll. 4-11, after portions of the template have been identified the HTML file is generated).

14. Regarding claim 32, Tiemann, Tso and Rana teach the method further comprising arranging the page components into the data model comprises arranging the page components into a hierarchical tree data model (Tiemann, col. 6, ll. 1-37, use of template file and example template file).

15. Regarding claim 33, Tiemann, Tso and Rana teach the method wherein generating the page component associated with each user control that supports output caching and is not available comprises:

retrieving the instructions that are associated with the page components from the respective separate file (Tso, col. 4, line 65 – col. 5, line 11); and

generating the page components based on the retrieved instructions (Tiemann, col. 7, ll. 4-11, after portions of the template have been identified the HTML file is generated).

16. Regarding claim 34, Tiemann, Tso and Rana teach the method further comprising: storing in the cache of the server computing device any generated page component that supports output caching and that is not available at the cache of the server computing device (Tso, col. 4, line 65 – col. 5, line 11).

17. Claims 19-22, 29, and 35-37 are rejected under 35 U.S.C. 103(a) as being unpatentable over Tiemann, Tso and Rana in view of Schloss et al. (US 6,249,844 B1), hereinafter referred to as Schloss.

18. Regarding claim 19, Tiemann and Tso teach the utilization of a cache as outlined above but do not explicitly recite wherein the output caching directive includes a time duration during which the component object is permitted to reside in the output cache. However, in related art, Schloss teaches on this method wherein Schloss teaches the utilization of standard cache management policies including duration (6, ll. 46-51). One of ordinary skill in the art at the time of the applicant's invention would have found it obvious to utilize caching rules like the ones described by Schloss in combination with the invention as taught by Tiemann and Tso. One of ordinary skill in the art would have been motivated to utilize standard cache management policies as is known in the art to maximize the efficiency of the cache.

19. Regarding claim 20, Tiemann, Tso and Schloss teach the method wherein the output caching directive includes an attribute indicating a condition for varying the component object to be stored in the output cache (Schloss, col. 6, ll. 46-51).

20. Regarding claim 21, Tiemann, Tso and Schloss teach the method wherein the attribute indicates that the component object is to be stored in the output cache according to a type of browser used by the client computing system (Tso, col. 6, ll. 11-14).

21. Regarding claim 22, Tiemann, Tso and Schloss teach the method wherein the attribute indicates that the component object is to be stored in the output cache according to values of at least one parameter (Schloss, col. 6, ll. 46-51).

22. Regarding claim 29, Tiemann and Tso teach the utilization of a cache as outlined above but do not explicitly recite wherein the output caching directive includes a time

duration during which the component object is permitted to reside in the output cache. However, in related art, Schloss teaches on this method wherein Schloss teaches the utilization of standard cache management policies including duration (6, ll. 46-51). One of ordinary skill in the art at the time of the applicant's invention would have found it obvious to utilize caching rules like the ones described by Schloss in combination with the invention as taught by Tiemann and Tso. One of ordinary skill in the art would have been motivated to utilize standard cache management policies as is known in the art to maximize the efficiency of the cache.

23. Regarding claim 35, Tiemann, Tso and Schloss teach the method wherein the output caching directive includes an attribute indicating a condition for varying the component object to be stored in the output cache (Schloss, col. 6, ll. 46-51).

24. Regarding claim 36, Tiemann, Tso and Schloss teach the method wherein the attribute indicates that the component object is to be stored in the output cache according to a type of browser used by the client computing system (Tso, col. 6, ll. 11-14).

25. Regarding claim 37, Tiemann, Tso and Schloss teach the method wherein the attribute indicates that the component object is to be stored in the output cache according to values of at least one parameter (Schloss, col. 6, ll. 46-51).

26. Claims 23, 24, 26, 38 and 39 are rejected under 35 U.S.C. 103(a) as being unpatentable over Tiemann, Tso, Rana and Schloss in view of Mattson (U.S. Patent Number 5,434,992), hereinafter referred to as Mattson.

27. Regarding claim 23, Tiemann, Tso and Schloss teach the need to increase data output performance, but are silent on the use of performance counters to monitor output-caching performance. However, Mattson teaches the use of counters to measure the performance of a cache (col. 9, line 56 – col. 10, line 2). One of ordinary skill in the art at the time of the applicant's invention would have recognized the advantage of using performance counters in order to improve the output of data (Schloss, col. 2, lines 27-30). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to combine the performance counters disclosed by Mattson with the data output method using data caching disclosed by Schloss.

28. Regarding claim 24, Tiemann, Tso and Schloss teach the need to increase data output performance, but are silent on the use of hit and miss counters to monitor output-caching performance. However, Mattson teaches the uses of hit and miss counters to measure the performance of a cache (col. 9, lines 56-64).

29. Regarding claim 26, Tiemann, Tso and Schloss teach the need to increase data output performance, but is silent on the use of calculating an output cache hit ratio to monitor output-caching performance. However, Mattson teaches the use of calculating hit ratios in order to measure the performance of a cache (col. 9, lines 56-64).

30. Regarding claim 38, Tiemann, Tso and Schloss teach the need to increase data output performance, but are silent on the use of performance counters to monitor output-caching performance. However, Mattson teaches the use of counters to measure the performance of a cache (col. 9, line 56 – col. 10, line 2). One of ordinary

skill in the art at the time of the applicant's invention would have recognized the advantage of using performance counters in order to improve the output of data (Schloss, col. 2, lines 27-30). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to combine the performance counters disclosed by Mattson with the data output method using data caching disclosed by Schloss.

31. Regarding claim 39, Tiemann, Tso and Schloss teach the need to increase data output performance, but are silent on the use of hit and miss counters to monitor output-caching performance. However, Mattson teaches the uses of hit and miss counters to measure the performance of a cache (col. 9, lines 56-64).

32. Claims 25 and 40 are rejected under 35 U.S.C. 103(a) as being unpatentable over Tiemann, Tso, Rana and Schloss in view of Smith et al (U.S. Patent Number 5,802,600), hereinafter referred to as Smith.

33. Regarding claim 25, Tiemann, Tso and Schloss teach the need to increase data output performance, but are silent on counting the number of additions and removals to the output cache. However, Smith taught gathering statistics based on directory entries to measure output-caching performance (col. 5, lines 8-54). One of ordinary skill in the art at the time of the applicant's invention would have recognized the advantage of using performance counters in order to improve the output of data (Schloss, col. 2, lines 27-30). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to combine the statistics gathering

method disclosed by Smith with the data output method using data caching disclosed by Schloss.

34. Regarding claim 40, Tiemann, Tso and Schloss teach the need to increase data output performance, but are silent on counting the number of additions and removals to the output cache. However, Smith taught gathering statistics based on directory entries to measure output-caching performance (col. 5, lines 8-54). One of ordinary skill in the art at the time of the applicant's invention would have recognized the advantage of using performance counters in order to improve the output of data (Schloss, col. 2, lines 27-30). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to combine the statistics gathering method disclosed by Smith with the data output method using data caching disclosed by Schloss.

Response to Arguments

35. Applicant's arguments filed 03 July 2008 have been fully considered but they are not persuasive.

36. With respect to the rejection of claim 14 under 35 USC 103(a) in view of Tiemann and Tso as set forth above, applicant argues (A) that the cited art, specifically Tiemann, does not teach "the component object cached using a cache key that comprises an identifier that is unique for each occurrence of the component object."

37. (A) With respect to this argument, the examiner respectfully disagrees. Tiemann teaches the utilization of a caching system wherein the contents of static portions are stored in appropriate database files (col. 7, ll. 41-51). For accuracy purposes, a version

check is performed to determine that the static portion being utilized is the most recent portion made available for use. Tiemann teaches in column 7, lines 60-67 the utilization of unique identification of cached objects wherein a filename of a static HTML file is generated based on the filename of a template and the timestamp associated with the template. Therefore, this unique identification utilizing template filename and time stamps teaches the component object that is identified by a unique key.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Nickerson (US 2008/0177859 A1) teaches a system and method for displaying dynamic page content in a page-caching browser.

Zimowski (US 2008/0155056 A1) teaches a technique for maintaining and managing dynamic web pages stored in a system cache and referenced objects cached in other data stores.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Benjamin Ailes whose telephone number is (571)272-3899. The examiner can normally be reached Monday-Friday, 5:30-8:30AM, 1:00-6:00PM, IFP Hoteling schedule.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Andrew Caldwell can be reached on 571-272-3868. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/B. A. A./
Examiner, Art Unit 2442

/Andrew Caldwell/
Supervisory Patent Examiner, Art Unit 2442